Digital filters for Hot-Mix Asphalt Complex Modulus Test Data Using Genetic Algorithm Strategies

Madhav V. Chitturi, Anshu Manik, and Kasthurirangan Gopalakrishnan

Abstract**—**The dynamic or complex modulus test is considered to be a mechanistically based laboratory test to reliably characterize the strength and load-resistance of Hot-Mix Asphalt (HMA) mixes used in the construction of roads. The most common observation is that the data collected from these tests are often noisy and somewhat non-sinusoidal. This hampers accurate analysis of the data to obtain engineering insight. The goal of the work presented in this paper is to develop and compare automated evolutionary computational techniques to filter test noise in the collection of data for the HMA complex modulus test. The results showed that the Covariance Matrix Adaptation-Evolutionary Strategy (CMA-ES) approach is computationally efficient for filtering data obtained from the HMA complex modulus test.

Keywords**—**HMA, dynamic modulus, GA, evolutionary computation.

I. INTRODUCTION

THE goal of the work presented in this paper is to develop and compare automated evolutionary computational techniques to filter test noise in the collection of data for the Hot-Mix Asphalt (HMA) complex modulus test. The dynamic or complex modulus (E^*) test is considered to be a mechanistically based laboratory test to reliably characterize the strength and load-resistance of HMA mixes [1]. A typical dense-graded HMA mixture is made of 86 percent by volume of aggregates bound with about 10 percent by volume of asphalt and incorporates about 4 percent of air-voids. The newly released Mechanistic-Empirical Pavement Design Guide (MEPDG) uses the dynamic modulus test to characterize HMA mixes used on interstate highways and most other high-volume highways that require superior load resistance [2].

For linear visco-elastic materials such as HMA mixtures, the stress-strain relationship under a continuous sinusoidal loading is defined by its complex dynamic modulus (E^*) [3]. This is a complex number that relates stress to strain for linear visco-elastic materials subjected to continuously applied sinusoidal loading in the frequency domain.

The HMA complex modulus is defined as the ratio of the amplitude of the sinusoidal stress at any given time, t, and the angular load frequency, ω, $\delta = \delta_0 \sin(\omega t)$ and the amplitude of the sinusoidal strain $\varepsilon = \varepsilon_0 \sin(\omega t - ø)$, at the same time and frequency, that results in a steady state response (Fig. 1 [4]):

$$E^* = \frac{\delta}{\varepsilon} = \frac{\delta_0 e^{i\omega t}}{\varepsilon_0 e^{i(\omega t-\phi)}} = \frac{\delta_0 \sin \omega t}{\varepsilon_0 \sin(\omega t - \phi)} \tag{1}$$

where, δ_0 = peak (maximum) stress; ε_0 = peak (maximum) strain; ø = phase angle, degrees; ω = angular velocity; t = time, seconds; i = imaginary component of the complex modulus

Mathematically, the dynamic modulus is defined as the absolute value of the complex modulus:

$$\left|E^*\right| = \frac{\delta_0}{\varepsilon_0} \tag{2}$$

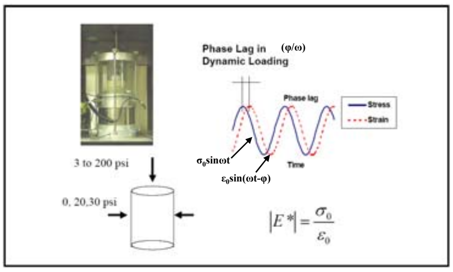


Fig. 1 HMA dynamic (complex) modulus test [4]

Complex modulus laboratory testing of asphalt concrete mixtures typically involves the use of cylindrical specimens of 150 mm height and 100 mm diameter subjected to simulated sinusoidal loading at strain levels targeted to be in the linear visco-elastic range [5]. In the laboratory, a sinusoidal (haversine) axial compressive stress is applied to a cylindrical asphalt concrete specimen at a given temperature and loading frequency. The applied stress and the resulting recoverable axial strain response of the specimen is measured and used to calculate the complex modulus (E^*) and phase angle (ø).

Given the complex visco-elastic nature of the asphalt

Manuscript received November 7, 2006.

Mr. Madhav V. Chitturi is with University of Illinois at Urbana-Champaign, Urbana, IL 61801 USA (phone: 217-333-8988; fax: 217-333-1924; e-mail: chitturi@uiuc.edu).

Dr. Anshu Manik is with the Department of Civil Engineering, University of Illinois at Urbana-Champaign, Urbana, IL 61801 USA (e-mail: manik@uiuc.edu).

Dr. Kasthurirangan Gopalakrishnan is with the Department of Civil Engineering, Iowa State University, Ames, IA 50011 UAA (e-mail: rangan@iastate.edu).

mixture, the temperatures and frequencies for the test methods pose an appreciable challenge for test equipment and measurement transducers. Test machine instrumentation issues can result in errors greater than 10% in the measurement of dynamic modulus for the HMA mixtures [6]. The most common observation is that the data collected from these tests are often noisy and somewhat non-sinusoidal (see Fig. 2) [7]. This hampers accurate analysis of the data to obtain engineering insight.

The solution to the presence of noise in the data calls for an appropriately designed filter that would eliminate the frequency components corresponding to the noise from the signal. There exist several standard ways to do this, but each suffer from certain limitations.

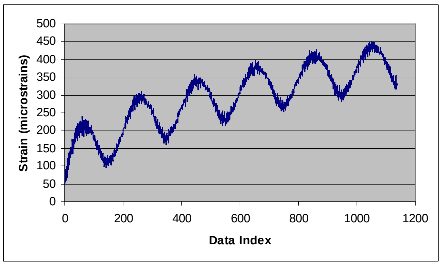


Fig. 2 Typical strain response from (HMA dynamic (complex) modulus test

II. MOTIVATION

During the HMA complex modulus laboratory testing, a fully computer-controlled digital data acquisition system records the time history of the applied load and the axial deformations. The load is measured with an electronic load cell in contact with one of the specimen caps. Axial deformations are measured with linear variable differential transformers. Strain gauges, or more often extensometers, affixed at various positions on the test specimens measure strain, and the data are automatically fed into a computer to calculate test results. The non-homogenous nature of HMA mix contributes to noise in the test data. Further, the data acquisition devices have limitations on their accuracy and resolution. As the strain response data has components of electronic noise, it is often required to filter the data before using it for analysis.

Hardware-level filters might have been a possible solution to this problem. However, hardware-level filters, especially at higher test frequencies, could easily cause attenuation of desired frequency responses. There are also standard digital filters available for this purpose. For instance, the Chebyshev Type II filter and the Butterworth filter [8] are frequently used in this context. But all these filters need to be designed specifically for each data set to extract optimal performance. Commercial tools like the signal processing toolbox provided with MATLAB® enable optimal filter design for specific sets of data. However, to automate this process the use of Genetic Algorithms (GAs) could be very helpful.

Calculation of HMA dynamic modulus involves the use of peak-to-peak amplitude of the stress (loading) and strain (response) [5]. To automate the analysis of data obtained from these tests, a peak search algorithm is generally required to determine the magnitudes of peaks and troughs and hence amplitude in the stress and strain records.

III. BACKGROUND

Genetic Algorithms (GAs) have been used in a large number of applications in the field of signal processing [9]. Tang et al. [10] describe some of the most successful applications of GAs in the signal processing area. These include IIR Adaptive Filtering, Nonlinear Model Selection, Active Noise Control and Speech Processing applications. Standard procedures are available to design Finite Impulse Response (FIR) filters automatically without using genetic algorithms [11-15]. However, such traditional optimizing filter design techniques suffer from various problems as pointed out by Suckley [16] who first applied GAs to digital filter design specifically to the design of low-pass FIR filters.

GAs provide methodologies that is automated, rapid, and gives filter realizations near minimal computational complexity. Roberts and Wade [17] extended Suckley's [16] work to design medium-order multiplier-less FIR filters. Dexiang et al. [18] used a parallel GA to design optimal Finite Word Length (FWL) FIR filters. Gentili et al. [19] used genetic algorithms for design of digital FIR filters with coefficients constrained to be sums of power-of-two terms.

Most of such algorithms require knowledge of desired frequency and / or amplitude output from the filter. However, this is difficult to obtain for the problem under consideration, as the data is sampled at different sampling rates and tests are run at different frequencies as well. This is done so because of the trade-off between having larger amount of data within the limitations of the data acquisition system, and increased computation time. Therefore there is a need for filter design which can function optimally without a priori knowledge of the test conditions.

In the problem under consideration, the only available data is the strain response in the form of text input of strain measurement values from all the measurement channels. Most of the algorithms discussed above require sophisticated knowledge of filtering and/or signal processing fundamentals to be able to use them. The laboratory technician collecting the HMA complex modulus test data may not have such a technical background, which is frequently the case.

However, as the Genetic Algorithm is a blind search algorithm, it does away with the need for specialized knowledge of the application area, if implemented appropriately. The unique contribution of this paper is that the methodology developed herein filters noise from the HMA modulus test data even in the absence of any knowledge of the desired signal output and can be used without specialized domain knowledge of filtering. Thus, it has the potential to be implemented commercially for rapid, real-time, and routine analysis of HMA complex modulus (E^*) test data.

IV. APPROACHES TO FILTER DESIGN

A one-dimensional digital filter function Y = FILTER (B, A, X) filters the data in vector X with the filter described by vectors A and B to create the filtered data Y. The filter is a "Direct Form II Transposed" implementation of the standard difference equation:

$$a(1)y(n) = b(1)x(n) + b(2)x(n-1) + ... + b(nb+1)x(n-nb) - a(2)y(n-1) - ... - a(na+1)y(n-na)$$

Thus, filter design involves the determination of the coefficients of the two vectors A and B, which can be done using several methods.

MATLAB has standard functions for designing digital filters given the parameter values. The Chebyshev type II low pass filter can be designed using the function ‘cheby2’. This function takes the following three parameters as input:

1. *Filter order (N):* it should be between 3 and 18. It has been found that higher order filters do not produce good results with data obtained from E* tests.
2. *Stop band ripple (R):* it should be between 1 and 64 decibels.
3. *Pass band edge frequency (Wn):* theoretically it should be between 0.0 and 1.0.

The function ‘cheby2’ returns the filter coefficients in length N+1 vectors B (numerator) and A (denominator). GA was used to optimize the suitability of the filter for any given data set by choosing the best possible combination of above-mentioned parameter values. Two different GA strategies were applied to do this. The goal was to determine which strategy would provide the most efficient and robust method for optimization. The methods applied were:

1. Binary coded genetic algorithm using mutation and crossover
2. Real coded GA with mutation and crossover
3. 1 + 1 evolution strategy
4. Evolution Strategy with Covariance Matrix Adaptation [22,23]

The first approach relies on the use of standard functions available in MATLAB®. However, to be able to design filters without the necessity of MATLAB would involve determining filter coefficients that cheby2 function calculates. The second approach, therefore, focuses on determining optimal coefficients for the multipliers in the numerator (B) and denominator (A) of a filter function. However, the number of these coefficients in any filter depends on the order of the filter. All the above mentioned four GA strategies were also tested with this approach.

V. FUNCTION EVALUATION AND FITNESS FUNCTION

The first step in function evaluation involves designing a filter with given parameter values. Then the designed filter is applied to a given E* test data set. Then the fitness of the filtered test data is evaluated.

The objective of filtering is to get a smooth response. At the same time the output should be as close to the measured response as possible, otherwise the filtering process may introduce attenuation which amounts to loss of information. Several criteria for measuring smoothness were considered. But in order to achieve robustness over variegated types of data, the criterion that was selected was a function of the total number of instances where the slope changes sign (which are called as peaks). Therefore, it is desirable to have the same number of peaks in the filtered data as the sinusoidal cycles of loading that the specimen was subjected to. Lesser number of peaks would indicate attenuation as a result of over-filtering. As the number of peaks goes below the optimal value the filtered data would be farther away from the original data. But the number of peaks also would vary for different data sets. The actual deviation of the output response from original data is the other criterion in this problem. Therefore, this is a case of multi-objective optimization.

To account for the multiple objectives it was decided that the two objectives namely, number of peaks and deviation (of the filtered data form the original data) be combined with appropriate weights to define fitness value. Criterion for deviation was calculated by computing the l2-norm, the original data being the reference.

There is a slight change in the fitness function used with different strategies which is discussed later. This had to be done because different strategies responded more favorably to slightly different fitnesses.

VI. DESIGNING THE CHEBYSHEV TYPE II FIR LOW PASS FILTER

As discussed previously, the Chebyshev function requires three parameters to generate the coefficients of a filter: order, ripple and edge band frequency. Initially order was fixed, and ripple and edge band frequency were the only parameters to be optimized.

TABLE I
GA PARAMETERS FOR BINARY-ENCODING FOR 2 INPUT PARAMETERS TO THE CHEBYSHEV FUNCTION

Selection	Proportionate (s= 2 and with replacement), Tournament
Crossover	Single point-crossover, p = 0.60
Mutation	p = 0.01
Len. of chromosome	20
Population size	100
No. of generations	100

This was because the inclusion of order as a parameter makes the problem much more complex as the order determines the number of variables to be optimized itself. But the quality of filtering would vary with order. Therefore, order was also included as a parameter to be optimized. The length of each chromosome was 20 in the case of binary-coded GA (Table I).

The GA with order as a variable was implemented as a real-coded GA as well. The length of each chromosome was 3 (Table II). Mutation was accomplished by generating a normal random number with a mean of zero for all the parameters. The standard deviations for order, ripple and edge band frequency were 1, 5 and 0.10 respectively.

TABLE II
GA PARAMETERS FOR BINARY-ENCODING FOR 3 INPUT PARAMETERS TO THE CHEBYSHEV FUNCTION

Selection	Proportionate (s= 2 and with replacement), Tournament
Crossover	Single point-crossover, p = 0.60
Mutation	p = 0.01
Len. of chromosome	3
Population size	100
No. of generations	100

Since this problem required a chromosome of size three only it was expected that evolutionary strategies may also work. The advantage with evolutionary strategies is that they need much fewer function evaluations as compared to binary or real GA. The 1 + 1 evolutionary strategy (Table III) worked well for problems with fixed order, but failed to converge when order also became an input parameter.

The implementation of the Covariance Matrix Adaptation-Evolutionary Strategy (CMA-ES) has led to significant advancements in efficiency and robustness in many applications. In many cases CMA-ES has been found to give linear time complexity. Another advantage of ES is that they can be easily parallelized [21]. CMA-ES can also reliably adapt to an arbitrarily oriented scaling of the search space [23]. Therefore, CMA-ES approach was well-suited to the particular problem at hand. For more details the reader is encouraged to read the cited papers.

VII. DESIGNING A GENERIC FILTER IMPLEMENTED AS A FIR LOW PASS FILTER

As stated earlier generic digital filters are a set of coefficients that are used to process noisy data and get smoothened output. These filters are characterized by order, the number of terms for numerator and denominator, and the coefficients themselves. Initially a binary simple GA code was written, which would optimize the parameters for a fixed order of the filter. That is the GA would only optimize the filter coefficients. In the next phase, order of the filter was also introduced as a parameter to be optimized. Since the number of filter coefficients, depends on the order of the filter, the length of each chromosome is not fixed. Rather it depends on the order, which is itself a parameter to be optimized by the GA. The order could vary from 3 to 18. Therefore, the number of filter coefficients could vary from 5 to 28.

TABLE III
GA PARAMETERS FOR BINARY-ENCODING FOR 1 + 1 ES ENCODING WITH 2 INPUT PARAMETERS TO THE CHEBYSHEV FUNCTION

Selection	Proportionate (s= 2 and with replacement), Tournament
Crossover	Single point-crossover, p = 0.60
Mutation	p = 0.01
Len. of chromosome	3
Population size	100
No. of generations	100

The initiation module first generates the order of the individual, and then based on the order, generates the required number of coefficients for each individual. Using single-point crossover between individuals of different lengths could yield invalid individuals. Therefore for solving this problem, two-point crossover was implemented, with both the points of crossover being less than the length of the shorter of the two parents. This would ensure that the crossover would not yield invalid (in terms of chromosome length) individuals. Mutation was also performed once the crossover operation was performed. While performing both mutation and crossover, it was ensured that the order is not varied. However this should not be a serious limitation. Because the population size is 100 and the possible number of orders is only 16, each order would be represented in the initial population.

An accuracy of three decimal places requires that each of the coefficients be represented by 10 bits and the order be represented by 4 bits (for a range of 3 to 8). Therefore in the binary-coded GA, the length of the chromosome could vary from 54 to 284 bits.

In real-coded GA the length of the chromosome could vary from 6 to 29. In the case of real-coded GA the mutation was accomplished by generating a normal random number with a mean of zero and standard deviation of 0.10.

CMA-ES strategy was applied to this problem as well. Default values as described in the last section were used for the parameters.

VIII. DISCUSSION OF RESULTS

Results for running all the algorithms on two typical data files are discussed in this section. These two files differ in the magnitude of the strain and sampling rate used while performing the dynamic modulus test. File 1 has around 2000 data points and corresponds to a high sampling rate while, File 2 has nearly 600 data points and corresponds to a medium sampling rate. For each data file the actual noisy data in shown in gray and the cleaned data in black. The bottom plot in each figure shows the mean fitness for each generation in the case of binary-coded GA, real-coded GA while in the case of CMA-ES approach it shows the max fitness in each generation. Only selected results are presented in this section due to space constraints.

Figs. 3, 4 and 5 display the results obtained using the binary-coded GA, real-coded GA (for the sake of brevity only the results for tournament selection are presented) and the CMA-ES approach for designing Chebyshev type II Low Pass Filter.

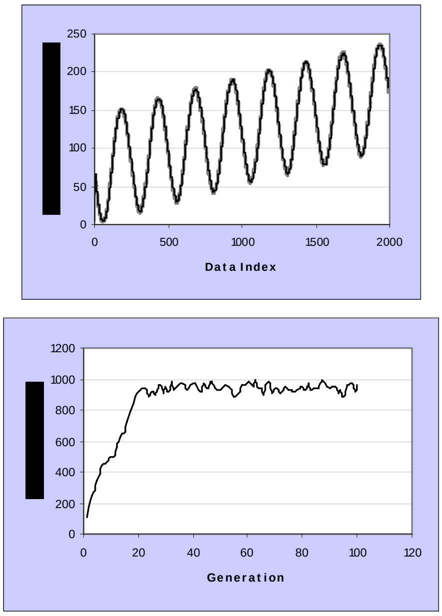
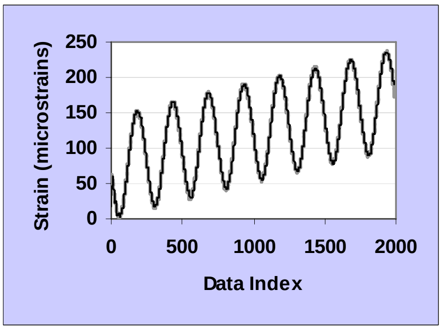


Fig. 3 Results obtained from binary-coded GA for Chebyshev Type II FIR Filter for File 1



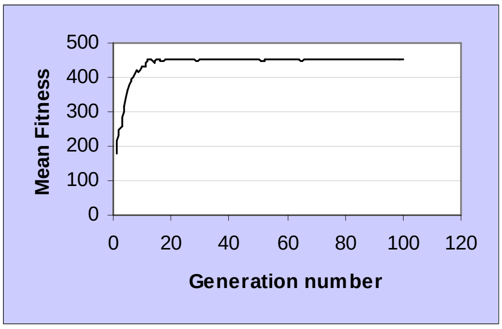


Fig. 4 Results obtained from real-coded GA for Chebyshev Type II FIR Filter for File 1

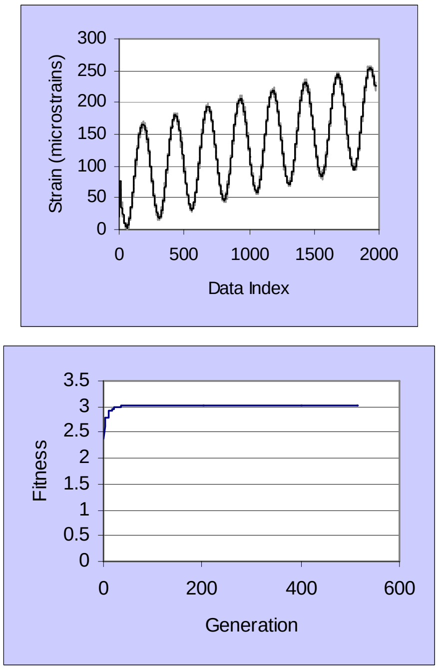


Fig. 5 Results obtained from CMA-ES approach for Chebyshev Type II FIR Filter for File 1

The actual fitness values of the end results are not compared because the weights used for deviation and peaks in the fitness function were slightly different depending on the data file and also the approach used.

It can be seen that maximum strain in data file 1, reaches 240 microstrain while in data file 2, it only reaches 115 microstrain. But the ideal number of peaks in both the files is the same. Naturally, different weights had to be used for deviation in the fitness function evaluation of the two files. Also in the CMA-ES approach it was found that obtaining average deviation by dividing total deviation using $n^{0.3}$ instead of n resulted in better convergence. This is because the deviations are maximum at the crests and troughs and are less in all the other points. Therefore averaging it over all the points resulted in underestimating the deviation. Nevertheless, it can be seen from Figs. 3 through 5 that all the three approaches resulted in similarly clean data. Also all of them reached pretty much the stable fitness in approximately 20 generations. However, the number of fitness evaluations performed differs significantly. In the case of binary and real-coded GA, 100 function evaluations are performed in each generation. While in the case of CMA-ES approach only 4 (λ=4) function evaluations are performed in each generation. Therefore the CMA-ES approach, took the least time for converging to the solution.

These approaches were tested on more data files as well. It was found that the CMA-ES approach was more consistent in cleaning the different data files. This also showed that the CMA-ES approach is not only more efficient but also more robust than the binary-coded and real-coded approaches when used for designing Chebyshev type II Low Pass Filter.

Fig. 6 shows the results obtained using binary-coded GA for designing a generic Low Pass FIR Filter. Due to space constraints, the results for other approaches could not be presented.

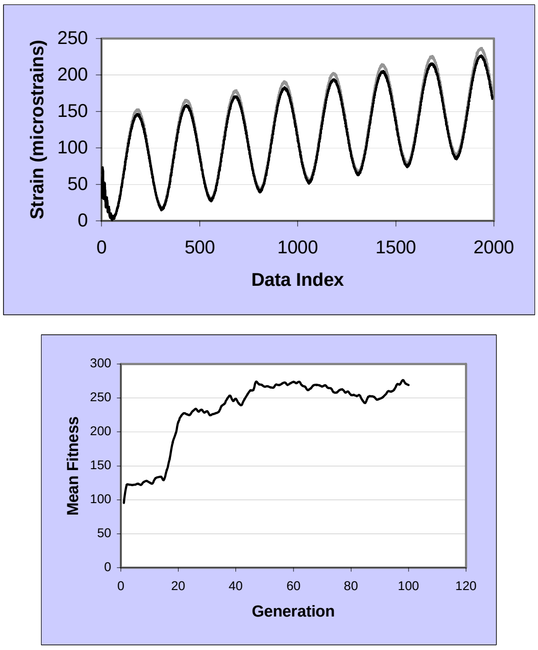


Fig. 6 Results obtained from binary-coded GA for generic Low Pass Filter for File 1

In the case of Chebyshev filter all the three approaches resulted in similar results. But in the case of generic low pass filter, it was found that only CMA-ES approach converged to good results. Both the binary-coded and real-coded approaches performed poorly and inconsistently. The binary-coded GA gave reasonably good results for data file 1 but the deviation is very significant in the case of data file 2. In the case of data file 2 for real-coded GA it was run for 200 generations because the mean fitness did not stabilize in 100 generations. Nevertheless the deviation was significant. Thus, the deviation in the results of the real-coded GA is high in both the cases.

It should be noted that in the case of generic filter design, the chromosome lengths are variable. In the case of real-coded and binary-coded GA, when the crossover occurs between chromosomes representing two completely different orders, the alleles exchanged most likely would represent completely different parameters and possible different number of parameters. Therefore crossover amongst different orders might be the reason for the poor performance of real-coded and binary-coded GA.

A better approach would have been to evolve individuals of each order separately, for say, n generations and then select the population for the next generation based on fitness. Then once again, segregate the individuals based on order and let them evolve independently for another n generations. Due to lack of time, this approach was not implemented. However, the authors believe that if this approach were implemented both the real-coded and binary-coded GA would perform better.

Also in the case of real-coded GA a single step size was used for all the parameters. This might have caused further deterioration in the performance of the real-coded GA. The CMA-ES approach adapts the step size and so it would be expected to perform better than these two approaches.

A four-layer feedforward network consists of a set of sensory units (source nodes) that constitute the input layer, two hidden layer of computation nodes, and an output layer of computation nodes. The following notation is generally used to refer to a particular type of architecture that has two hidden layers: (# inputs)-(# hidden neurons)-(# hidden neurons)-(# outputs). For example, the notation 10-40-40-3 refers to an ANN architecture that takes in 10 inputs (features), has 2 hidden layers consisting of 40 neurons each, and produces 3 outputs.

IX. CONCLUSION

The dynamic or complex modulus (E^*) test is considered to be a mechanistically based laboratory test to reliably characterize the strength and load-resistance of Hot-Mix Asphalt (HMA) mixes used in the construction of roads. The goal of the work presented in this paper is to develop and compare automated evolutionary computational techniques to filter test noise in the collection of data for the HMA complex modulus test.

In this study, different Genetic Algorithm (GA) strategies for designing digital filters for HMA complex modulus test data were evaluated. Two types of digital filter were considered: Chebyshev type II FIR low pass filter and the generic filter implemented as a FIR low pass filter.

For Chebyshev type II Low Pass filter, the binary-coded, real-coded GA and the Covariance Matrix Adaptation-Evolutionary Strategy (CMA-ES) approach resulted in similar results. However, the CMA-ES approach was more robust and computationally efficient. In the case of generic Low Pass filter, only CMA-ES approach converged to good results. Possible reasons for the poor performance of binary-coded and real-coded GAs were stated and some solutions to possibly overcome them have been suggested. The CMA-ES approach was also found to be computationally most efficient in the case of generic filter design. The study demonstrated that GAs are indeed useful approaches to filtering data obtained from the HMA complex modulus test.

REFERENCES

[1] T. Harman, “Using the Dynamic Modulus Test to Assess the Mix Strength of HMA,” *Public Roads* 64 (6), 2001

[2] NCHRP, “Guide for Mechanistic-Empirical Design of New and Rehabilitated Pavement Structures,” *NCHRP Project 1-37A*, Transportation Research Board, Washington, D.C., 2004.

[3] C.E. Dougan, J.E. Stephens, J. Mahoney, and G. Hansen. “E^* - Dynamic Modulus. Test Protocol – Problems and Solutions,” *Report No. CT-SPR-0003084-F-03-3*, Connecticut Department of Transportation, Rocky Hill, Connecticut, 2003.

[4] C.W. Schwartz, “ENCE 447: Pavement Engineering.” Unpublished Course Material - Spring 2005, Department of Civil Engineering. University of Maryland, College Park, Maryland, 2005.

[5] NCHRP 1-37, “A Draft Test Method A1. Standard test method for dynamic modulus of asphalt concrete mixtures and master curves,” Arizona State University, Tempe, AZ, 2000.

[6] J.S. Daniel, G.R. Chehab, and Y.R. Kim. “Issues Affecting Measurement of the Complex Modulus of Asphalt Concrete,” *Journal of Materials in Civil Engineering* 16(5): 469-476, 2004.

[7] T. Pellinen, “Comparison of analysis techniques to obtain modulus and phase angle from sinusoidal test data,” *Proc. 6th International RILEM Symposium on Performance Testing and Evaluation of Bituminous Material*, PTEBM ’03, Zurich, Switzerland, 2003.

[8] R. Levy and S.B. Cohn, “A History of Microwave Filter Research, Design, and Development,” *IEEE Transactions on Microwave Theory and Techniques*, Vol. MIT-32, No. 9, 1984.

[9] D.E. Goldberg, *Genetic Algorithms in search, optimization and machine learning*, Addison-Wesley, 1989.

[10] K.S. Tang, K.F. Man, S. Kwong and Q. He, “Genetic Algorithms and their applications,” *IEEE Signal Processing Magazine*, 13(6), 1996.

[11] J.H. McClellan and T.W. Parks, “A unified approach to the design of optimum FIR linear-phase digital filters,” *IEEE Tran. Circuit theory*, CT-20: 697–701, 1973.

[12] J.W. Adams and A.N. Jr. Wilson, “A new approach to FIR digital filters with fewer multipliers and reduced sensitivity,” *IEEE Trans. Circuits and Systems*, CAS-30: 277–283, 1983.

[13] J.W. Adams and A.N. Jr. Wilson, “Some efficient digital pre-filter structures,” *IEEE Tran.*, CAS-31: 260–265, 1984.

[14] Y. Neuvo, D. Cheng-Yu, and S.K. Mitra, “Interpolated finite impulse response filters,” *IEEE Trans. Acoustics, Speech, and Signal Processing*, ASSP-32: 563–570, 1984.

[15] G. Wade, P. Van-Eetvelt, and H. Darwen, ”Synthesis of efficient low-order FIR filters from primitive sections,” *IEE Proc. G*, 137: 70-87, 1984.

[16] D. Suckley, “Genetic algorithm in the design of FIR,” *IEE Proceedings, Part G: Electronic Circuits & Systems*, 138(2): 234-238, 1991.

[17] A. Roberts and G. Wade, “A structured GA for FIR filter design,” *Proc. workshop on Natural Algorithms in Signal Processing*, Vol. 1, Chelmsford, UK, 1993.

[18] X.J. Dexiang and M.L. Delay, “Design of Finite Word Length digital filters using a parallel genetic algorithm,” *Proc. IEEE Conference*, pp. 834–837, 1992.

[19] P. Gentili, F. Piazza, and A. Uncini, “Improved power-of-two sharpening filter design by genetic algorithm,” ICASSP, *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, Vol. 3, Piscataway, NJ, pp. 1375–1377, 1996.

[20] MATLAB. MATLAB® online help. MathWorks, Inc., 2004.

[21] H.P. Schwefel, *Evolution and Optimum Seeking*, Sixth Generation Computer Technology Series, New York: John Wiley and Sons Inc., 1995.

[22] N. Hansen and A. Ostermeier, “Completely Derandomized Self-Adaptation in Evolution Strategies,” *Evolutionary Computation*, 9(2): 159–195, 2001.

[23] N. Hansen, S. D. Müller, and P. Koumoutsakos, “Reducing the Time Complexity of the Derandomized Evolution Strategy with Covariance Matrix Adaptation (CMA-ES),” *Evolutionary Computation*, 11(1): 1–18, 2003.